

# **Telemetrie pro dálkově řízené modely**

## **Telemetry System for RC Models**

## Zadání bakalářské práce

Student:

**Miroslav Beleš**

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R059 Mobilní technologie

Téma:

Telemetrie pro dálkově řízené modely  
Telemetry System for RC Models

Zásady pro vypracování:

Cílem bakalářské práce je navrhnout a realizovat hardwarový modul pro vzdálený přenos telemetrických dat. Přenos dat bude probíhat v ISM pásmu pomocí některého z dostupných rádiových modulů. Připojení k počítači bude realizováno pomocí přijímače připojeného na USB nebo sériový port. Pro modul vysílače bude navržen vhodný komunikační protokol a ten implementován v dostupném jednočipovém počítači (Atmel, Microchip apod.)

- 1) Rešerše telemetrických systémů a ISM modulů, jejich vzájemné srovnání.
- 2) Návrh komunikačního protokolu pro odesílání dat, zabezpečení přenosu proti chybám.
- 3) Implementace komunikačních rutin do vybraného jednočipového počítače.
- 4) Implementace přijímacího a dekódovacího software pro stolní počítač.
- 5) Otestování navrženého řešení (spolehlivost a dosah).

Seznam doporučené odborné literatury:

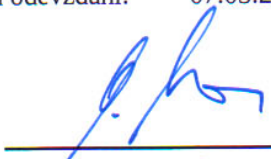
- [1] Paul Scherz, Practical Electronics for Inventors, Tab Electronics; 3 edition, 2013, ISBN 978-0071771337
- [2] Frank Carden, Telemetry Systems Engineering, Artech House; 2nd edition edition, 2002, ISBN 978-1580532570
- [3] Vladimír Váňa, Atmel AVR: Programování v jazyce C, BEN-Technická literatura, 2003, ISBN 80-7300-102-0

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.


Vedoucí bakalářské práce: **Mgr. Ing. Michal Krumník, Ph.D.**

Datum zadání: 01.09.2014

Datum odevzdání: 07.05.2015

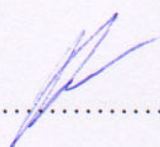
  
doc. Ing. Miroslav Vozňák, Ph.D.  
vedoucí katedry



  
prof. RNDr. Václav Snášel, CSc.  
děkan fakulty

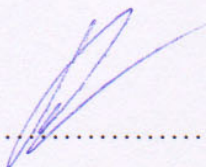
Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 *Studijního a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava*.

V Ostravě 7. května 2015

.....

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 7. května 2015

.....

Tímto bych chtěl poděkovat především Mgr. Ing. Michalu Krumníkovi, Ph.D. za odborné vedení práce, veškerý čas, věcné připomínky a rady.

## **Abstrakt**

Práce se zabývá problematikou na poli telemetrie určené pro dálkově řízené modely, s využitím mikropočítače, modulu pro přenos dat a senzoru pro měření telemetrických dat.

Praktická část je tvořena návrhem a realizací hardwarového telemetrického modulu s implementovanými komunikačními rutinami a dekodovacím softwarem pro počítač.

Teoretická část se zabývá popisem a porovnáním, v současné době populárních, bezdrátových modulů pro přenos dat a výběrem jednoho modulu pro realizaci hardwarového telemetrického modulu. Dále potom popisem celkového postupu realizace modulu a implementace dekodovacího softwaru.

**Klíčová slova:** Telemetrie, dálkově řízené modely, hardwarový telemetrický modul, dekodovací software, Arduino Nano, ESP8266, BMP180

## **Abstract**

This thesis is about problems on field of telemetry for radio-controlled models with usage of microcontroller, data transmission module and sensor for telemetry data measurement.

In practical part the hardware telemetry module is designed and implemented with implemented communication routines and decoding software.

Theoretical part deals with description a comparison of popular wireless modules for data transmission and choice of one of the described modules for implementation of hardware telemetry module. In this part is also described overall procedure of realisation of module and implementing decoding software.

**Keywords:** Telemetry, radio-controlled models, hardware telemetry module, decoding software, Arduino Nano, ESP8266, BMP180

## Seznam použitých zkratk a symbolů

AP	– Access Point
BSS	– Basic Service Set
CR	– Carriage Return
CRC	– Cyclic Redundancy Check
CSV	– Comma-separated values
FCFS	– First Come, First Served
FIFO	– First In, First Out
IEEE	– Institute of Electrical and Electronics Engineers
ISM	– Industrial, Scientific and Medical
JTAG	– Joint Test Action Group
LF	– Line Feed
MAC	– Media Access Control
P2P	– Peer-To-Peer
RAM	– Random Access Memory
RC	– Resistor, Capacitor
RC	– Radio Control
RF	– Radio Frequency
ROM	– Read Only Memory
RTC	– Real Time Clock
RX	– Receiver
SPI	– Serial Peripheral Interface
SRAM	– Static Random Access Memory
TCP/IP	– Transmission Control Protocol / Internet Protocol
TX	– Transmitter
UART	– Universal Asynchronous Receiver/Transmitter
WiFi	– Wireless Fidelity
WLAN	– Wireless Local Area Network

## Obsah

<b>1</b>	<b>Úvod</b>	<b>4</b>
<b>2</b>	<b>Porovnání bezdrátových modulů</b>	<b>5</b>
2.1	Modul ESP8266 . . . . .	5
2.2	Modul nRF24L01+ . . . . .	7
2.3	Modul CC1101 . . . . .	11
<b>3</b>	<b>Shrnutí bezdrátových modulů</b>	<b>14</b>
<b>4</b>	<b>Stavba telemetrického modulu</b>	<b>15</b>
4.1	Použité komponenty . . . . .	15
4.2	Schéma zapojení . . . . .	16
4.3	Práce s modulem ESP8266 . . . . .	17
<b>5</b>	<b>Implementace webového serveru</b>	<b>19</b>
5.1	Inicializace . . . . .	20
5.2	Přenos dat . . . . .	21
5.3	Použití telemetrického modulu . . . . .	21
<b>6</b>	<b>Implementace webového serveru a dekodovacího softwaru</b>	<b>24</b>
6.1	Implementace webového serveru s XML obsahem . . . . .	25
6.2	Implementace dekodovacího softwaru . . . . .	26
<b>7</b>	<b>Testování</b>	<b>30</b>
7.1	Průběh testování . . . . .	31
7.2	Test stability . . . . .	31
7.3	Výsledky testování . . . . .	32
<b>8</b>	<b>Závěr</b>	<b>33</b>
<b>9</b>	<b>Reference</b>	<b>34</b>

## Seznam tabulek

2.1	Odběr proudu nRF24L01+ ve standby mode . . . . .	9
2.2	Odběr proudu nRF24L01+ v RX mode . . . . .	9
2.3	Odběr proudu nRF24L01+ v TX mode . . . . .	9
2.4	Formát Enhanced ShockBurst™ paketu (nRF24L01+) . . . . .	10
2.5	Odběr proudu CC1101 v power down mode . . . . .	12
2.6	Odběr proudu CC1101 při RX/TX na frekvenci 315MHz . . . . .	12
2.7	Odběr proudu CC1101 při RX/TX na frekvenci 433MHz . . . . .	12
2.8	Odběr proudu CC1101 při RX/TX na frekvenci 868/915MHz . . . . .	13
3.1	Srovnání pracovních frekvencí modulů . . . . .	14
3.2	Srovnání odběru proudu bezdrátových modulů . . . . .	14
4.1	Typy AT příkazů . . . . .	17
6.1	Ukázka výstupní tabulky . . . . .	29
7.1	Parametry kvadrokoptéry . . . . .	30



## Seznam obrázků

2.1	ESP8266 . . . . .	5
2.2	ESP8266 - Piny . . . . .	6
2.3	nRF24L01+ . . . . .	8
2.4	nRF24L01+ - Piny . . . . .	8
2.5	CC1101 . . . . .	11
2.6	CC1101 - Piny . . . . .	11
4.1	Schéma zapojení telemetrického modulu . . . . .	16
5.1	Blokové schéma webového serveru . . . . .	19
5.2	Výstup v internetovém prohlížeči . . . . .	22
6.1	Blokové schéma přenosu XML dat . . . . .	24
6.2	Uživatelské rozhraní dekodovacího softwaru . . . . .	28
7.1	Kvadroptéra s telemetrickým modulem . . . . .	31
7.2	Měření teploty a tlaku . . . . .	32

# 1 Úvod

Dálkově řízené modely, především pak kvadrokoptéry, jsou stále populárnější, jak v amatérském modelářství, tak v oblasti složek integrovaného záchranného systému nebo třeba v mobilní skupině státního ústavu radiální ochrany. Pro tyto účely se dnes vyvíjí telemetrické moduly z důvodu okamžité zpětné vazby z dálkově řízeného modelu či pro pozdější analýzu letových údajů. Bezpilotní letouny s telemetrickým systémem se využívají pro měření všude tam, kde jsou podmínky pro člověka životu nebezpečné.

V této práci se zabývám návrhem a realizací telemetrického modulu, který dokáže komunikovat s implementovaným dekodovacím softwarem, ale i s obyčejným internetovým prohlížečem, bez nutnosti instalace jakéhokoli dodatečného softwaru. S použitím telemetrického modulu a internetového prohlížeče se můj telemetrický systém liší od ostatních na současném trhu. Tento telemetrický systém dokáže zpracovávat teplotu, atmosférický tlak, nadmořskou výšku a výšku nad úrovní země.

Hlavní součástí pro přenos telemetrických dat je použití jednoho z bezdrátových modulů. Popisem, porovnáním a výběrem vhodného bezdrátového modulu se zabývám v kapitolách č. 2 a 3. Po vybrání vhodného bezdrátového modulu je čas na stavbu samotného telemetrického modulu za použití mikropočítače a senzoru pro měření telemetrických dat. Celkový návrh a realizace telemetrického modulu je popsán v kapitole č. 4.

Aby bylo možné číst a zpracovávat telemetrická data, je nutné implementovat webový server a dekodovací software, který umožňuje nejen zobrazení telemetrických dat, ale také jejich zpracování v podobě CSV tabulky. Touto problematikou se zabývám v kapitolách č. 5 a 6.

V neposlední řadě je nutné hotový telemetrický systém otestovat v ostrém provozu. Jak tento test proběhl a jak telemetrický modul obstál v provozu je napsáno v kapitole č. 7.

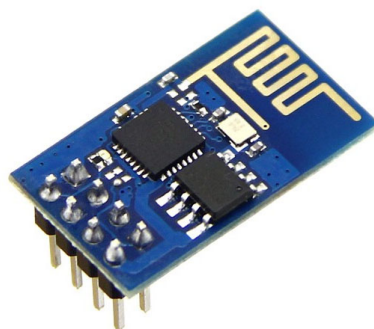
## 2 Porovnání bezdrátových modulů

V dnešní době je na trhu k dispozici široká škála různých modulů pro bezdrátový přenos dat. Vybral jsem a detailněji rozebral ty, které považuji za populární a dobře dostupné. Pro lepší čitelnost a přesnost, jsem některé klíčové názvy nepřekládal.

Jedním z hlavních kritérií pro výběr bezdrátového modulu je frekvence, na které dokáží jednotlivé moduly pracovat. Využívá se zejména bezlicenčního pásma ISM, které je využíváno v průmyslových, vědeckých a zdravotnických oborech. Odtud také vychází název pásma - Industrial, Scientific and Medical. Jelikož se jedná o bezlicenční pásmo, tedy je dovoleno v tomto pásmu používat homologovaných zařízení bez licenčních poplatků, není garantovaný přenos bez rušení. ISM pásmo je využíváno samozřejmě i pro komerční použití, například technologie WiFi i Bluetooth pracují právě v tomto pásmu. O hojné využití ISM pásma se zaslужují také radioamatéři a RC modeláři.

### 2.1 Modul ESP8266

Modul ESP8266 (na obrázku č. 2.1) je poměrně nový modul, uvedený na trh v roce 2014 firmou Espressif Systems. Pořizovací cena je v řádech několika desítek korun, je tudíž levný a dostupný téměř všem. Pracuje v bezlicenčním pásmu ISM na frekvenci 2,4 GHz na WiFi standardu IEEE 802.11 b/g/n. Hned po uvedení na trh se modul stal velmi populární a to hlavně díky jeho nízké ceně a univerzálnímu použití. Za velmi krátkou dobu kolem tohoto modulu vznikla velká komunita vývojářů. ESP8266 poskytuje dostupný způsob k realizaci domácí automatizace, přenosu telemetrie, spotřební elektroniky, IP kamer, „chytrých“ elektrických zásuvek a podobně. Modul ESP8266 je konfigurován pomocí AT příkazů, díky nimž je práce s modulem rychlá a bezchybná [1].



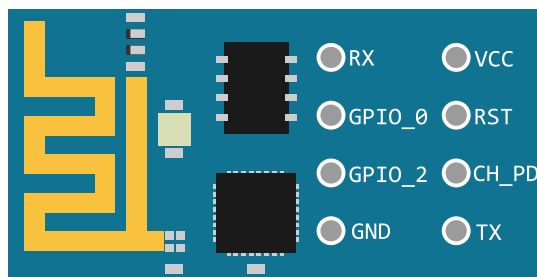
Obrázek 2.1: Bezdrátový modul ESP8266 <sup>1</sup>

---

<sup>1</sup>Zdroj: <http://hackaday.io/project/2879-ESP8266-WiFi-Module-Library>

### 2.1.1 Piny a jejich popis

Modul ESP8266 obsahuje celkem osm pinů v matici 2x4. Bohužel modul není příliš vstřícný k použití pro klasické nepájivé pole (otvory v rastru 2,4 mm), proto je třeba si koupit nebo vyrobit patici s širší roztečí pinů anebo modul zapojit klasicky pomocí kabelů samec-samice, které jsou kompatibilní s 2,4 mm roztečí. Na obrázku č. 2.2 je vidět výpis pinů.



Obrázek 2.2: Piny modulu ESP8266

### 2.1.2 Napájení a řízení spotřeby

Modul je nutno napájet maximálně 3,3 V, vyšší napětí by mohlo zařízení poškodit nebo zcela zničit. ESP8266 disponuje oproti ostatním bezdrátovým modulům vyšší spotřebou, maximálně 215 mA. Spotřeba je řízena ve třech režimech:

#### Active mode

V active modu modul spotřebovává nejvíce elektrické energie, všechny funkce čipu jsou v aktivním stavu, čekající na příkazy.

#### Sleep mode

Narozdíl od active mode, ve sleep mode logicky spotřebuje modul méně energie - odběr proudu je menší než 12  $\mu$ A. Ve sleep mode zůstávají v provozu pouze hodiny reálného času (RTC) a watchdog. Hodiny reálného času pak mohou být naprogramovány tak, aby byl ESP8266 vzbuzen v určité době.

#### Deep sleep mode

Deep sleep mode je třetí stupeň operačního režimu. V tomto režimu ESP8266 spotřebovává nejméně elektrické energie. Aktivní jsou pouze hodiny reálného času, zbytek funkcí je vypnut.

### 2.1.3 Procesor a paměť

Čip ESP8266 disponuje vestavěným 32-bitovým mikroprocesorem. K procesoru se dá přistupovat pomocí těchto rozhraní:

- iBus rozhraní, pro přístup k externí flash paměti
- dBus rozhraní pro přístup k řadiči paměti
- AHB rozhraní pro přístup k registrům
- JTAG rozhraní pro ladění (debugging)

ESP8266 používá k práci ROM a SRAM paměť. Přístup k nim zajišťuje rozhraní iBus, dBus a AHB. Kterékoliv s těchto tří rozhraní může požádat o přístup k paměti ROM a SRAM, paměťový řadič pak obslouží tyto požadavky na bázi FCFS (obsluha podle pořadí přístupu - kdo dřív přijde, bude dřív obsloužen) [2].

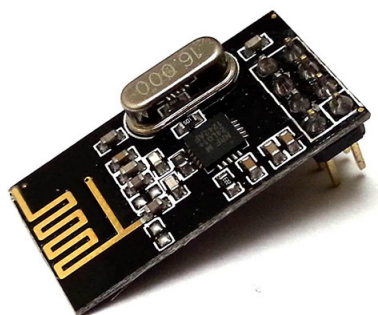
### 2.1.4 Firmware a rádio

Aplikace a firmware jsou nahrány z flash paměti při probuzení modulu, ty se pak zpracují v pamětech ROM a SRAM, které jsou integrovány na čipu. Firmware implementuje protokoly TCP/IP a IEEE 802.11 b/g/n WLAN MAC protokol. ESP8266 může pracovat na 14-ti kanálech od 2412MHz až po 2484MHz. Každý kanál je od sebe oddělen frekvencí 5MHz, kromě posledního (14-tého) kanálu, který je od předchozího (13-tého) oddělen frekvencí 12MHz [2].

K dispozici je pro ESP8266 velké množství firmwaru, ať už oficiálních, tak i neoficiálních, vyvíjených komunitou. Jednotlivé firmwary se od sebe liší použitými AT příkazy či celkovým vyladěním pro modul.

## 2.2 Modul nRF24L01+

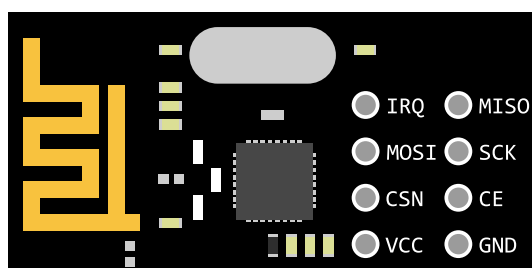
Čip nRF24L01+ (na obrázku 2.3), vyroben firmou Nordic Semiconductor je vysoce integrovaný čip, používaný jako vysílač dat pro pásmo ISM na frekvenci 2,4GHz. Výrobce uvádí, že jeho čip je vhodný pro počítačové periferie, herní průmysl, sport a fitness, hračky s spotřební elektronikou [3].



Obrázek 2.3: Bezdrátový modul nRF24L01+ <sup>2</sup>

### 2.2.1 Piny a jejich popis

Modul nRF24L01+ obsahuje, podobně jako ESP8266, matici pinů 2x4, tudíž celkem 8 pinů. Ani tento modul není spřátelený s nepajivým polem - řady pinů jsou příliš blízko u sebe. Na obrázku č. 2.4 je vidět výpis pinů.



Obrázek 2.4: Piny modulu nRF24L01+

### 2.2.2 Napájení a řízení spotřeby

Spotřeba se mění spolu se režimem modulu. Režimy dělíme do čtyř skupin - power down mode, standby modes, RX mode (přijímač) a TX mode (vysílač). Tyto režimy se dále dělí do určitých stavů. Každý stav má svá specifika a každý má jiný odběr proudu. Odběr proudu ve vypnutém stavu je  $I_{vdd\_pd} = 900\text{ nA}$ . V tabulkách č. 2.1, 2.2, 2.3 je vyobrazeno porovnání odběru proudu pro různé stavy a různé režimy modulu [4].

<sup>2</sup>Zdroj: <http://robino.ru/product/a1>

**Standby modes**

<b>Veličina</b>	<b>Stav</b>	<b>Hodnota</b>
$I_{vdd\_st1}$	Standby mode - I	26 $\mu$ A
$I_{vdd\_st2}$	Standby mode - II	320 $\mu$ A

Tabulka 2.1: Odběr proudu ve standby mode

**RX mode**

<b>Veličina</b>	<b>Stav</b>	<b>Hodnota</b>
$I_{vdd\_2m}$	Při rychlosti 2Mb/s	13,5 mA
$I_{vdd\_1m}$	Při rychlosti 1Mb/s	13,1 mA
$I_{vdd\_250}$	Při rychlosti 250kb/s	12,6 mA
$I_{vdd\_rxs}$	Průměrný odběr proudu při nastavování RX	8,9 mA

Tabulka 2.2: Odběr proudu v RX mode

**TX mode**

<b>Veličina</b>	<b>Stav</b>	<b>Hodnota</b>
$I_{vdd\_tx0}$	Při síle signálu 0 dBm	11,3 mA
$I_{vdd\_tx6}$	Při síle signálu –6 dBm	9 mA
$I_{vdd\_tx12}$	Při síle signálu –12 dBm	7,5 mA
$I_{vdd\_tx18}$	Při síle signálu –18 dBm	7 mA
$I_{vdd\_txs}$	Průměrný odběr proudu při nastavování TX	8 mA

Tabulka 2.3: Odběr proudu v TX mode

**2.2.3 Operační režimy**

Jak už jsem uvedl výše, nRF24L01+ má čtyři hlavní operační režimy. V následujícím textu je detailněji popíšu.

**Power down**

V tomto režimu, jak již napovídá, je nRF24L01+ deaktivován a má minimální odběr proudu. Všechny hodnoty uložené v registrech jsou uchovány a SPI je v aktivním stavu. Odběr proudu v tomto režimu je 900 nA.

## Standby modes

### Standby mode - I

Je určen pro minimalizování průměrného odběru proudu při zachování krátké doby spouštění. Pouze část krystalového oscilátoru je v tomto režimu aktivní. Přejít do/z aktivního režimu (RX/TX) záleží na nastavení pinu CE.

### Standby mode - II

V druhém pohotovostním režimu je odběr proudu vyšší než v prvním pohotovostním režimu. Je navíc aktivní dodatečná vyrovnávací paměť.

## RX mode

RX mode je považován za aktivní režim, kde je modul nRF24L01+ použit jako přijímač. Přijímač demoduluje signál z RF kanálu a demodulovaná data pak prezentuje protokolu v základním pásmu.

## TX mode

TX mode je také považován za aktivní režim, kde je modul nRF24L01+ vysílač. V režimu TX nRF24L01+ zůstává, dokud nejsou odeslány všechny pakety a po odeslání všech paketů se nRF24L01+ přepne do pohotovostního režimu - I.

### 2.2.4 Enhanced ShockBurst™

Enhanced ShockBurst™ je spojovací vrstva, která umožňuje automatické sestavení a časování paketů, automatické potvrzení a přeposílání paketů. Umožňuje realizaci vysoce výkonné komunikace s velmi nízkou spotřebou a malými náklady na výrobu.

Enhanced ShockBurst™ používá ke své činnosti ShockBurst™ pro automatickou manipulaci a časování paketů. ShockBurst™ při odesílání sestaví pakety a připraví data pro přenos. Naopak při přijímání hledá platnou adresu v demodulovaném signálu. Při nalezení platné adresy se paket zpracuje a potvrdí pomocí kontrolního součtu CRC. Pokud je paket platný, jsou jeho data přenesena do prázdného slotu FIFO paměti přijímače.

Formát Enhanced ShockBurst™ paketu obsahuje preambuli, adresu, pole pro řízení paketu, vlastní data a kontrolní součet. Vyobrazení paketu je vidět v tabulce č. 2.4 [4].

Preamble	Adresa	Pole pro řízení paketu	Data	CRC
1 byte	3 - 5 bytů	9 bitů	0 - 32 bytů	1 - 2 byty

Tabulka 2.4: Formát Enhanced ShockBurst™ paketu



## 2.3 Modul CC1101

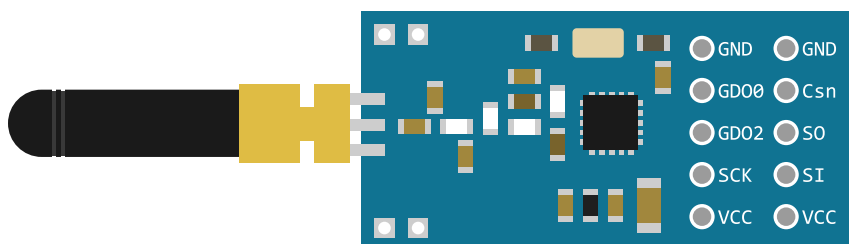
CC1101 (na obrázku 2.5) je vysílací/přijímací bezdrátový modul, vyrobený firmou Texas Instruments. Patří k méně používaným modulům své kategorie, avšak i tento modul si našel své příznivce. Dokáže pracovat v ISM pásmu, přesněji na frekvencích 315, 433, 868, a 915 MHz. Výrobce uvádí, že se hodí především do systémů s nízkou spotřebou, jako bezdrátové zvonky, bezdrátové senzory, na domácí automatizaci, ale i pro průmyslové monitorování a řízení [6].



Obrázek 2.5: Bezdrátový modul CC1101 <sup>3</sup>

### 2.3.1 Piny a jejich popis

Modul CC1101 obsahuje narozdíl od ESP8266 a nRF24L01+ celkem 10 pinů - matice 2x5. Jak je již u těchto modulů zvykem, je třeba pro zapojení do klasického nepájivého pole použít kabely samec-samice, kompatibilní s roztečí 2,4 mm nebo si koupit/vyrobít vhodnou patici pro zapojení. Výpis pinů je možné vidět na obrázku č. 2.6.



Obrázek 2.6: Piny modulu CC1101

<sup>3</sup>Zdroj: [www.dx.com/cs/p/cc1101-wireless-module-w-external-antenna-blue-160898](http://www.dx.com/cs/p/cc1101-wireless-module-w-external-antenna-blue-160898)

### 2.3.2 Napájení a řízení spotřeby

Modul CC1101 je napájen napětím 3,3 V. Odběr proudu je velmi detailně popsán v oficiální dokumentaci. Pro zjednodušení jsem vybral nejdůležitější stavy a jejich hodnoty odběru proudu [5].

Odběr proudu se v aktivních režimech pohybuje v rozmezí 10 – 35 mA, v závislosti na konkrétním režimu TX nebo RX a jejich stavu. V tabulkách 2.5, 2.6, 2.7, 2.8, jsou vidět odběry proudu v nejdůležitějších režimech.

**Power down mode**

Odběr proudu	Stav
0,2 $\mu$ A	Veškeré moduly jsou vypnuty
165 $\mu$ A	Takzvaný stav XOFF - všechny moduly vypnuty, až na regulátor napětí

Tabulka 2.5: Odběr proudu v power down mode

**Odběr proudu při RX/TX na frekvenci 315 MHz**

Odběr proudu	Stav
14,3 mA	V RX modu při rychlosti 38,4 kBaud
16,5 mA	V RX modu při rychlosti 250 kBaud
12,3 mA	V TX modu při vysílacím výkonu –6 dBm
27,3 mA	V TX modu při vysílacím výkonu +10 dBm

Tabulka 2.6: Odběr proudu při RX/TX na frekvenci 315MHz

**Odběr proudu při RX/TX na frekvenci 433 MHz**

Odběr proudu	Stav
15 mA	V RX modu při rychlosti 38,4 kBaud
17,1 mA	V RX modu při rychlosti 250 kBaud
13,1 mA	V TX modu při vysílacím výkonu –6 dBm
29,2 mA	V TX modu při vysílacím výkonu +10 dBm

Tabulka 2.7: Odběr proudu při RX/TX na frekvenci 433MHz

**Odběr proudu při RX/TX na frekvenci 868/915 MHz**

<b>Odběr proudu</b>	<b>Stav</b>
14,6 mA	V RX modu při rychlosti 38,4 kBaud
16,9 mA	V RX modu při rychlosti 250 kBaud
16,4 mA	V TX modu (frekvence 868 MHz) při vysílacím výkonu -6 dBm
34,2 mA	V TX modu (frekvence 868 MHz) při vysílacím výkonu +12 dBm
17 mA	V TX modu (frekvence 915 MHz) při vysílacím výkonu -6 dBm
33,4 mA	V TX modu (frekvence 915 MHz) při vysílacím výkonu +11 dBm

Tabulka 2.8: Odběr proudu při RX/TX na frekvenci 868/915MHz

### 3 Shrnutí bezdrátových modulů

V kapitole č. 2 jsem popsal tři bezdrátové moduly, které jsem považoval za vhodné pro přenos telemetrických dat. V úvahu jsem musel brát frekvenční pásmo, ve kterém dokáží jednotlivé moduly pracovat (ISM pásmo bylo podmínkou), dále také vhodnost na základě odběru proudu, složitost implementace a cenu.

V tabulce 3.1 vidíme srovnání jednotlivých modulů na základě pracovního frekvenčního pásma. Všechny porovnávané moduly jsou vhodné pro bezdrátový přenos telemetrických dat v rámci České republiky, podle všeobecného oprávnění č. VO-R/10/05.2014-3, vydaného Českým telekomunikačním úřadem. Porovnávání podle ceny i cen modulů. Ceny jsou uvedeny ke dni 25. ledna 2015, jsou převzaty z internetového obchodu Dx.com a převedeny v kurzu \$1 = 24,86 Kč [8].

Název modulu	Pracovní frekvenční pásmo	Vhodnost	Cena
ESP8266	2,4 GHz	Vhodný	134,49 Kč
nRF24L01+	2,4 GHz	Vhodný	74,33 Kč
CC1101	315 , 433 , 868 a 915 MHz	Pouze v pásmu 433 MHz	246,85 Kč

Tabulka 3.1: Srovnání pracovních frekvencí modulů

Dalšímu srovnání jsem podrobil moduly na základě odběru proudu - tabulka č. 3.2. Toto srovnání je mírně zavádějící, jelikož každý z modulů používá rozdílné provozní režimy.

Název modulu	Odběr proudu v TX režimu
ESP8266	215 mA
nRF24L01+	11,3 mA
CC1101	13,1 mA

Tabulka 3.2: Srovnání odběru proudu bezdrátových modulů

Ke své práci na telemetrickém modulu pro dálkově ovládané modely jsem se rozhodl použít bezdrátový modul ESP8266, který na trh přišel během psaní této bakalářské práce. K tomuto rozhodnutí jsem dospěl zejména kvůli řízení modulu pomocí AT příkazů a velmi velké komunitě uživatelů<sup>4</sup>, což mi k mé práci poskytne dobrý základ pro kvalitní implementaci telemetrického modulu. Modul ESP8266 umožňuje implementaci kompletního WiFi řešení, což spolu s HTTP protokolem umožňuje bezchybnou komunikaci s klientským softwarem. Navíc modul ESP8266 poskytuje velmi dobrou a čitelnou dokumentaci, která vývojářům zjednoduší, zkvalitní a zrychlí práci.

<sup>4</sup>Komunita modulu ESP8266 má své fórum na adrese <http://www.esp8266.com/>

## 4 Stavba telemetrického modulu

Telemetrický modul je soubor čidel a senzorů, které snímají fyzikální vlastnosti určitého objektu, například aut, modelů, strojů a podobně. Fyzikálními vlastnostmi je myšlena teplota, tlak, sklon, rychlost, výška, gravitační zrychlení a mnoho dalších vlastností, které se liší podle snímaného objektu. V mém případě je snímán objekt RC model (konkrétně kvadrokoptéra) a snímané fyzikální vlastnosti jsou teplota okolí, atmosférický tlak, nadmořská výška a výška nad zemským povrchem.

Telemetrické moduly mají široké spektrum využití. Jsou často využívány v motoristickém sportu, ve vědeckých pokusech, v oblasti RC modelů, v různých průmyslových strojích a podobně.

K realizaci telemetrického modulu pro RC modely jsem využil dva způsoby implementace. První způsob zajišťuje komunikaci mezi telemetrickým modulem (na bázi webového serveru) a internetovým prohlížečem na adrese 192.168.4.1. Druhý způsob je realizován jako komunikace mezi telemetrickým modulem a dekódovacím softwarem instalovaným na počítači, kdy dekódovací software přistupuje k telemetrickému modulu na adrese 192.168.4.1/xml.

Oba způsoby komunikace jsou implementovány v jednom skeči (sketch) na mikropočítači Arduino Nano. Uživatel si tedy může vybrat způsob komunikace pro jeho potřeby. Například, pokud nemá uživatel po ruce notebook s dekódovacím softwarem, vystačí si s mobilním telefonem, na kterém se zobrazí aktuálně měřená telemetrická data v internetovém prohlížeči. V tomto se liší můj telemetrický modul od ostatních.

### 4.1 Použité komponenty

#### Arduino Nano v3.0

Jako hlavní řídicí jednotku jsem ke své práci použil mikropočítač Arduino Nano ve verzi v3.0. Tento mikropočítač je osazen mikroprocesorem **ATmega328** s 32kB pamětí a taktem 16 MHz. Arduino Nano disponuje čtrnácti digitálními vstupy/výstupy a osmi analogovými vstupy. Díky jeho obrovské komunitě a přijatelné ceně patří Arduino k velmi populárním mikropočítačům, které se dají využít v širokém spektru použití. Při rozměrech 45 mm x 18 mm a váze pouhých 5 g je tento mikropočítač ideálním kandidátem pro práci s RC modely, nebo všude tam, kde je zapotřebí malých rozměrů a nízkých hmotností [9].

#### BMP180

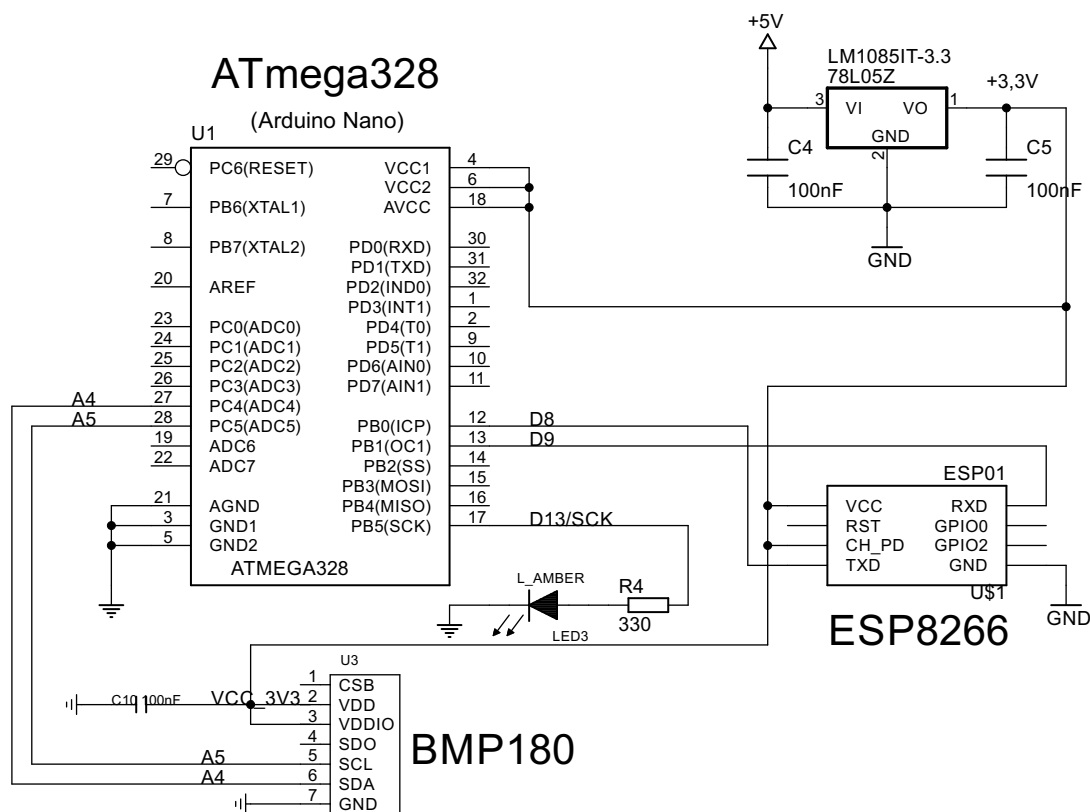
Pro čtení přenášených údajů jsem použil senzor BMP180 od firmy Bosch. Tento drobný a levný senzor dokáže snímat atmosférický tlak v rozsahu 300 – 1100 hPa. Mimo jiné obsahuje i digitální teploměr pracující v rozmezí od –40 do 85 °C. Při znalosti atmosférického tlaku a teploty není problém spočítat nadmořskou výšku. Senzor BMP180 tedy dokáže pracovat v nadmořské výšce přibližně od –500 do 9000 m nad mořem, což je pro účely telemetrického modulu pro oblast RC modelů více než dostačující [10].

## ESP8266

Jak již bylo zmíněno v kapitole č. 3, k realizaci přenosu telemetrických dat jsem použil modul ESP8266, který zajistí bezpečný a bezchybný přenos telemetrických dat.

## 4.2 Schéma zapojení

Na obrázku 4.1 je zobrazeno zapojení modulu ESP8266, senzoru BMP180 s mikropočítačem Arduino Nano. Modul ESP8266 je třeba napájet z externího zdroje, jelikož Arduino Nano dokáže poskytnout pouze 40 mA, což je pro běh modulu ESP8266 málo. Dále je třeba brát v potaz, že modul ESP8266, stejně jako senzor BMP180, dokáže pracovat pouze pod napětím 3,3 V, kdežto Arduino Nano musí být napájeno napětím 5 V. Z tohoto důvodu je využito stabilizátoru napětí LM1085IT-3.3, který z 5 V ze zdroje stabilizuje napětí na 3,3 V, vhodného pro napájení modulu ESP8266 a senzoru BMP180.



Obrázek 4.1: Schéma zapojení telemetrického modulu

Zapojení jako takové poté není příliš složité. Pin TX z ESP8266 jde na pin D9 mikropočítače, stejným stylem pak pin RX je připojen na pin D8 mikropočítače. Piny D8 a D9 jsou konfigurovány jako TX a RX mikropočítače, díky kterým může mikropočítač komunikovat s modulem ESP8266. Dále je patrné, že pin CH\_PD neboli chip select modulu ESP8266 je zapojen na logickou jedničku.

U teplotního a tlakového modulu BMP180, který má čtyři piny – SDA, SCL, GND, VIN, je zapojení následující. Pin SDA – datový kanál – je zapojen na vstup A4 mikropočítače. SCL – hodinový signál – pak směřuje na pin A5 mikropočítače.

### 4.3 Práce s modulem ESP8266

Jak již bylo řečeno v kapitole č. 2.1, modul ESP8266 je konfigurován pomocí jednoduchých AT příkazů. Modul je možno konfigurovat čtyřmi typy AT příkazů. V tabulce 4.1 vidíme jejich popis.

Typ	Formát	Popis
Testování	AT+<x>=?	Testování nastavovacích příkazů nebo dotazy ohledně rozsahu hodnot nastavovacího příkazu.
Dotazování	AT+<x>?	Vrací aktuální hodnotu parametru.
Nastavení	AT+<x>=<...>	Nastaví hodnotu parametru definovanou uživatelem.
Vykonání	AT+<x>	Spustí příkaz bez nastavení hodnot.

Tabulka 4.1: Typy AT příkazů

Všechny AT příkazy musí pro vykonání příkazu končit `\r\n` neboli CR+LF (nový řádek). Po odeslání AT příkazu do modulu ESP8266 se příkaz vykoná a modul odešle odpověď. Pokud příkaz proběhl úspěšně, je odpovědí jednoduché OK. Pokud se naopak v příkazu vyskytne chyba, je odpovědí ERROR, česky „chyba“. Když je modul připraven k činnosti (například po restartu příkazem AT+RST), vrací se odpověď `ready`, česky „připraven“. Jestliže vykonáme AT příkaz pro změnu parametru a tento parametr již je změněn na požadovanou hodnotu, vrací modul kladnou odpověď `no change`, česky „beze změny“ [11].

#### Jazyk C

Samotná komunikace mezi modulem ESP8266 a mikropočítačem je implementována v programovacím jazyce C. Jazyk C byl vyvinut už v 70. letech 20. století a dnes je využívám zejména pro psaní takzvaného systémového softwaru. Mnoho moderních programovacích jazyků „dělí“ syntaxi právě z jazyka C, mezi nimi i C# nebo Java.

Abychom mohli komunikovat s ESP8266, je třeba zajistit dvě sériové komunikace. Jednu pro informace o ladění a druhou pro komunikaci s ESP8266. Vzhledem k tomu, že Arduino Nano disponuje pouze jednou sériovou komunikační linku, je třeba vyhledat vhodnou alternativu. Je

možno využít buď knihovnu `SoftwareSerial`, která nám umožní přiřadit sériovou komunikaci i na jiné piny mikropočítače, nebo hardwarový převodník, který by nám umožnil externí sériový port neboli UART.

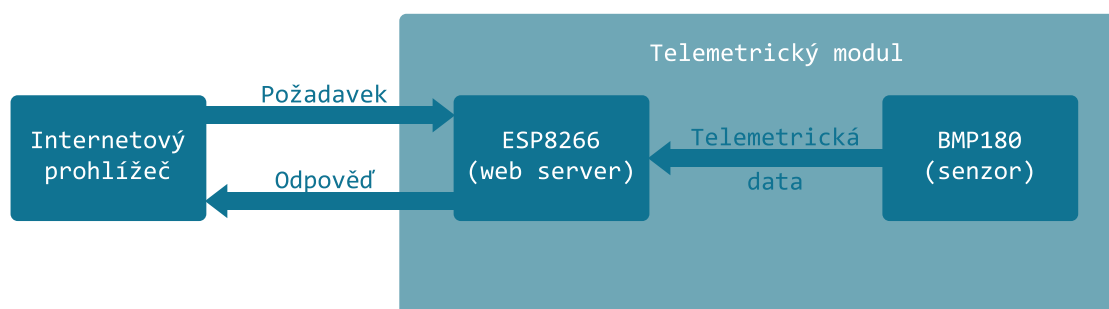
Rozhodl jsem se pro využití knihovny `SoftwareSerial`. Modul ESP8266 s poslední verzí firmwaru (v1.0.0) přichází se znakovou rychlostí 115 200 baud, rychlostí, kterou `SoftwareSerial` nezvládne obsluhovat. V této verzi firmwaru lze znakovou rychlost měnit pomocí AT příkazu `AT+UART`. Oproti minulým verzím firmwaru je tento příkaz nový. V nižších verzích šla znaková rychlost měnit příkazem `AT+BAUD`, tento příkaz se ovšem ve verzi v1.0.0 již nevyskytuje. Je tedy nutné znakovou rychlost snížit na přijatelnou hodnotu, v tomto případě na hodnotu 9 600 baudů.

Ještě před samotnou implementací přenosu telemetrických dat je vhodné změnit SSID přístupového bodu. Tento krok není nutný, ale určitě přispěje k pohodlí uživatele. Nastavení údajů o přístupovém bodu se může provést pouze jednou, není tedy nutné nastavovat tyto údaje při každé inicializaci modulu. Nejprve je vhodné modul restartovat příkazem `AT+RST`, poté je třeba nastavit modul jako přístupový bod, aby bylo možné nastavit jeho atributy. To se vykoná příkazem `AT+CWMODE=2`, kde číslo dvě značí, že bude modul ESP8266 fungovat jako přístupový bod. Dále už stačí jen použít příkaz pro nastavení atributů přístupového bodu `AT+CWSAP="Telemetry_Module", "vsbBEL0075", 3, 3`, kde „Telemetry\_Module“ značí nově nastavené SSID, „vsbBEL0075“ značí heslo pro přístup na WiFi síť, první číslo 3 znamená kanál, na kterém se bude vysílat a druhé číslo 3 znamená, že zabezpečení přístupového bodu je WPA2-PSK.



## 5 Implementace webového serveru

Cílem webové aplikace je zobrazení telemetrických dat ve webovém prohlížeči, bez nutnosti instalovat jakýkoliv specializovaný software. Myšlenkou je použít modul ESP8266 jako webový server, který by na vyžádání odeslal snímaná data do prohlížeče koncového uživatele.



Obrázek 5.1: Blokové schéma webového serveru

Na obrázku 5.1 je vidět blokové schéma komunikace mezi telemetrickým modulem a uživatelem. Uživatel odešle HTTP požadavek z internetového prohlížeče na webový server, tedy na ESP8266, tomu jsou za pomoci mikropočítače dodána telemetrická data, naměřená senzorem BMP180. Odpovědí modulu ESP8266 na HTTP požadavek je odpověď ve formě webové stránky, ve které jsou obsaženy naměřené údaje o teplotě, atmosférickém tlaku, nadmořské výšce a výšce nad povrchem země.

### Protokol HTTP

HTTP neboli Hyper Text Transfer Protocol je protokol pro přenos objektů libovolného typu mezi webovým serverem a internetovým prohlížečem. Pod pojmem libovolný typ si můžeme představit HTML stránky, obrázky nebo jakákoliv jiná data. Základním modelem HTTP protokolu je:

- Navázání spojení
- Zaslání požadavku na webový server klientem
- Zaslání odpovědi webového serveru klientovi
- Ukončení spojení

HTTP protokol používá ke komunikaci několik metod, mezi které patří od verze 1.0 metody GET, HEAD a POST. Od verze 1.1 se HTTP protokol rozšiřuje o metody PUT, DELETE a TRACE. Protokol HTTP pracuje na portu 80 a funguje nad protokolem TCP [13].

## Protokol TCP

Protokol TCP neboli Transmission Control Protocol pracuje na třetí vrstvě modelu OSI a jeho úkolem je dopravovat data mezi aplikacemi na různých počítačích. Protokol TCP garantuje spolehlivé doručení ve správném pořadí. TCP je spojově orientovaný protokol [14].

## HTML

HyperText Markup Language neboli HTML je jednoduchý značkovací jazyk určený pro tvorbu webových stránek. Celý zdrojový kód je reprezentován množinou tagů s jejich atributy. Pro přechody mezi jednotlivými HTML stránkami s využitím hypertextových odkazů.

## 5.1 Inicializace

První věcí, kterou je nutno provést, je nastavení sériových linek na vhodnou znakovou rychlost – v tomto případě na hodnotu 9 600 baud. To provedeme dvojicí příkazů `Serial.begin(9600)` a `esp8266.begin(9600)`, kde `Serial` je sériová linka pro ladění a `ESP8266` je dodatečně vytvořená sériová linka, pomocí knihovny `SoftwareSerial`, pro komunikaci mikropočítače s bezdrátovým modulem `ESP8266`. Před každým použitím je třeba bezdrátový modul restartovat, aby se modul řádně inicializoval. Další bod konfigurace, kterou je třeba udělat, je nastavení `ESP8266` jako přístupový bod, neboli AP. Poté už zbývá jen umožnit vícenásobný přístup k AP a nastavení samotného `ESP8266`, jako server na portu 80. Pro ladění je také vhodné nechat si zobrazit IP adresu bezdrátového modulu – přes tuto adresu budeme přistupovat k telemetrickému modulu v internetovém prohlížeči. Jako výchozí nastavená IP adresa AP je 192.168.4.1. Dalším krokem je zavolání inicializační funkce pro senzor `BMP180` pomocí příkazu `sensorBMP180.begin()`. Pro práci se senzorem `BMP180` byla použita knihovna `Adafruit_BMP085.h`, výrobce `Adafruit`<sup>5</sup>. Knihovna pro senzor `BMP085` je kompatibilní se senzorem `BMP180`. Po úspěšné inicializaci teplotního senzoru můžeme do proměnné uložit nadmořskou výšku, pomocí funkce `readAltitude()`, která je aktuální v době inicializace. S touto hodnotou budeme později pracovat na získání výšky nad zemským povrchem. Níže je uveden seznam příkazů, které byly použity k inicializaci modulu. Tyto příkazy jsou vykonány pouze jednou, při každém spuštění bezdrátového modulu.

### Použité AT příkazy k inicializaci modulu

- `AT+RST` - Restart modulu
- `AT+CWMODE=2` - Nastavení modulu jako AP (při `AT+CWMODE=1` by byl modul nastaven jako stanice, při `AT+MODE=3` pak jako AP i stanice)
- `AT+CIPMUX=1` - Nastavení vícenásobného přístupu k AP (hodnota 1 - povoleno, 0 - zakázáno)

<sup>5</sup>Zdroj: <https://github.com/adafruit/Adafruit-BMP085-Library>

- `AT+CIPSERVER=1,80` - Nastavení modulu jako server na portu 80 – HTTP port (příkaz ve formě `AT+CIPSEVER=X, Y`, kde `X` může nabývat dvou hodnot: 1 - vytvoření serveru, 0 - zrušení serveru. `Y` pak značí port, na kterém bude server provozován)
- `AT+CIFSR` - Odpovědí na tento příkaz je nastavená IP adresa modulu.

## 5.2 Přenos dat

Po úspěšné inicializaci modulu můžeme přejít k samotnému přenosu dat mezi webovým serverem a koncovým internetovým prohlížečem. Inicializace probíhá v bloku kódu, který se provede pouze jednou (funkce `void setup()`) – při spuštění programu v mikropočítači, kdežto přenos probíhá v bloku kódu, který probíhá v nekonečné smyčce (funkce `void loop()`).

Na počátku každé z iterací funkce `loop()` je vhodné přečíst aktuální hodnoty teploty, atmosférického tlaku, nadmořské výšky a výšky nad povrchem země. To lze provést jednoduše přes funkce:

- `readTemperature()` - funkce pro čtení teploty
- `readPressure()` - funkce pro čtení atmosférického tlaku
- `readAltitude()` - funkce pro čtení nadmořské výšky

Pro získání hodnoty výšky nad povrchem země je použito rozdílu aktuální nadmořské výšky a nadmořské výšky, která byla uložena při inicializaci telemetrického modulu.

Dalším krokem je hlídání, zda byl na modul zaslán požadavek na zobrazení dat. Požadavek je zaslán do modulu ESP8266, jestliže se uživatel snaží připojit na IP adresu 192.168.4.1 z webového prohlížeče. Pokud požadavek byl zaslán, bezdrátový modul si uloží ID aktuálního spojení mezi modulem a uživatelem. Dále se veškeré hodnoty získané ze senzoru BMP180 vhodně zformátují a uloží do řetězce, obsahujícího HTML hlavičku a text s telemetrickými daty. Modul musí být informován, kolik bytů bude zasláno. To se provede příkazem `AT+CIPSEND=X, Y`, kde `X` je ID spojení a `Y` je počet bytů k zaslání. Při jednom přenosu zvládne modul ESP8266 přenést maximálně 2048 bytů. Po vykonání příkazu `AT+CIPSEND` bude modul čekat na data k odeslání. Odeslání dat se provádí posláním zformátovaného řetězce na sériovou linku modulu ESP8266. Po přenosu je vhodné spojení uzavřít příkazem `AT+CIPCLOSE=X`, kde `X` je ID spojení, které má být ukončeno.

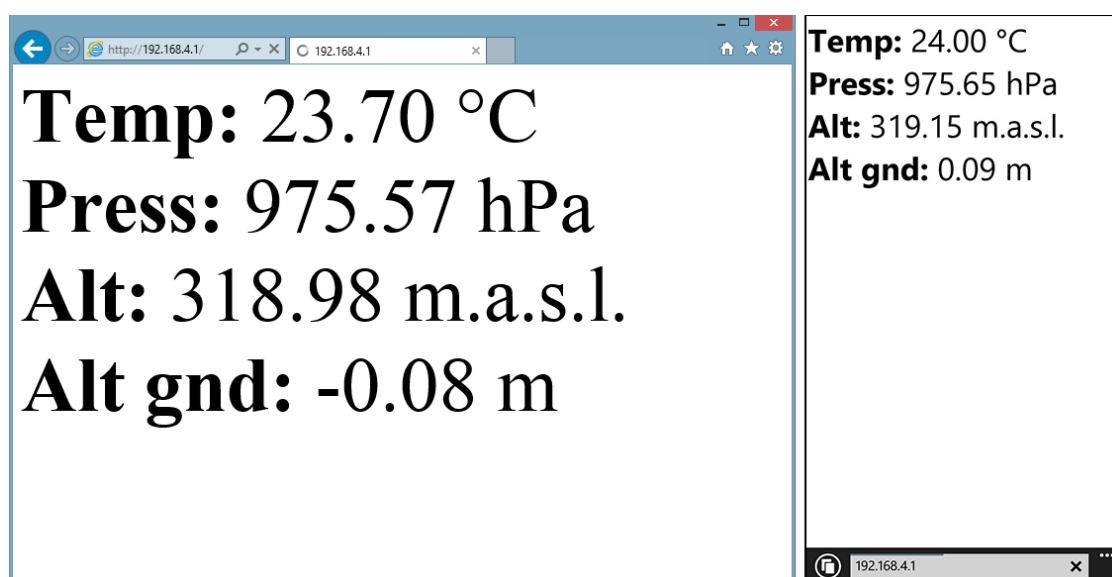
## 5.3 Použití telemetrického modulu

Používání telemetrického modulu implementovaného pomocí webového serveru je velmi jednoduché. Uživatel se musí připojit na WiFi síť s SSID „Telemetry\_Module“ a kvůli bezpečnosti vyplnit heslo - „vsbBEL0075“. Dále je třeba, aby do internetového prohlížeče uživatel vepsal IP adresu

modulu, která je 192.168.4.1. Po vykonání požadavku se v internetovém prohlížeči zobrazí HTML stránka s aktuálními údaji z telemetrického modulu. Dále se už uživatel nemusí o nic starat, díky HTML hlavičce je nastavena stránka tak, aby se po třech sekundách po načtení sama aktualizovala. Pokud již uživatel nechce ve sledování údajů pokračovat, může internetový prohlížeč uzavřít a odpojit se od WiFi sítě telemetrického modulu.

Ve volném prostoru a za ideálních povětrnostních podmínek má telemetrický modul v tomto provedení dosah až 430 m, přičemž v této vzdálenosti stále bezproblémově odesílá data k uživateli.

Na obrázku 5.2 jsou zobrazeny výstupní informace v internetovém prohlížeči. Jak je patrné z obrázku, může uživatel zobrazit data jak na klasickém počítači, tak i v mobilním zařízení.



Obrázek 5.2: Výstup v internetovém prohlížeči

Je také nutné dodat, že vzhledem k tomu, že se nadmořská výška propočítává pomocí teploty a atmosférického tlaku, není údaj o nadmořské výšce zcela přesný. Z toho logicky vyplývá, že ani výška nad zemským povrchem není zcela přesná, protože se vychází z rozdílu hodnoty aktuální nadmořské výšky a nadmořské výšky při inicializaci modulu. Je tedy nutné počítat s určitými odchylkami. V reálné situaci je přesnost nadmořské výšky a výšky nad povrchem země zhruba  $\pm 3$  m.

#### Výhody:

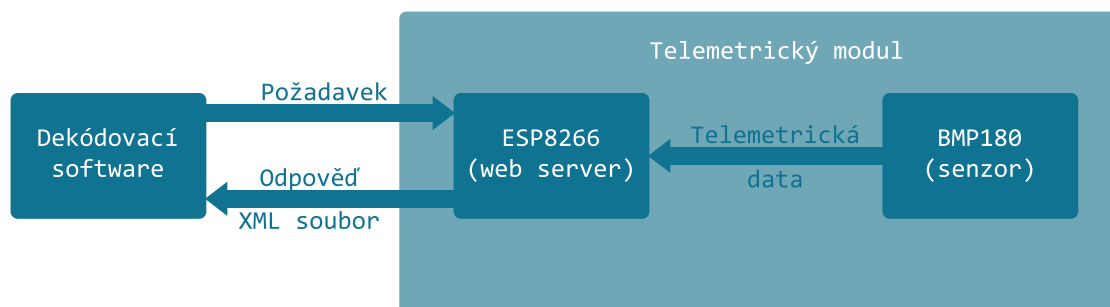
- Není nutnost instalovat žádný speciální software
- Zobrazení dat v jakémkoliv zařízení s webovým prohlížečem – zejména v mobilních telefonech

**Nevýhody:**

- Nemožnost ukládání dat
- Nemožnost práce s daty – grafy, tabulky a podobně
- Pomalejší přenos

## 6 Implementace webového serveru a dekódovacího softwaru

Cílem tohoto způsobu komunikace mezi telemetrickým modulem a počítačem je vytvoření webového serveru na modulu ESP8266, který obsahuje XML dokument s daty, která jsou připravena pro přenos. Vhodně implementovaný dekódovací software si v určeném intervalu vyžádá data z telemetrického modulu, která pak po přijetí z modulu ESP8266 prezentuje ve svém uživatelském rozhraní. S daty navíc implementovaný software umí dále pracovat.



Obrázek 6.1: Blokové schéma přenosu XML dat do dekódovacího softwaru

Na obrázku 6.1 je vidět blokové schéma přenosu mezi telemetrickým modulem a dekódovacím softwarem. Dekódovací software si vyžádá XML soubor z modulu ESP8266, který obsahuje telemetrická data přečtená ze senzoru BMP180. Odpovědí modulu ESP8266 je XML soubor, uložený na webovém serveru na adrese 192.168.4.1/xml, odkud si požadovaný XML soubor dekódovací software převezme a s předanými daty dále pracuje.

### Jazyk XML

Jazyk eXtensible Markup Language (XML) je značkovací jazyk. Rozdíl mezi XML a HTML je takový, že XML data popisuje, zatímco HTML data zobrazuje. Vlastní data jsou vepsána do párových tagů, které jsou pojmenované vhodným názvem, například `<jmeno>Karel</jmeno>`. Jednotlivé tagy se do sebe dají vnořit, podobně jako u HTML [15].

XML soubor, který je přenášen z modulu ESP8266 do klientského softwaru je poměrně jednoduchý. Root elementem je `<telemetry>`, který v sobě uchovává veškerá měřená data. Celý XML dokument je vyobrazen na výpisu 6.1.

---

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <telemetry>
3   <temp>23.70</temp>
4   <pressure>978.34</pressure>
5   <altitude>294.45</altitude>
6   <ground>0.77</ground>
7 </telemetry>
```

---

Výpis 6.1: XML dokument s telemetrickými daty

## 6.1 Implementace webového serveru s XML obsahem

Inicializace webového serveru je popsána již v kapitole č. 5.1 – oba dva způsoby komunikace pracují na stejném softwaru nahaném v mikropočítači. V tomto případě je ovšem rozdíl v tom, že cílem není zobrazení výsledků měření v internetovém prohlížeči, nýbrž jejich distribuce do dekodovacího softwaru.

Pokud se bude chtít spojit dekodovací software s vytvořeným webovým serverem, bude na požadavek ke spojení vyžadovat vhodnou odpověď v podobě HTTP hlavičky. Pokud se HTTP hlavička nezašle, skončí dekodovací software s chybou narušení HTTP protokolu. Abych se ujistil, že se HTTP hlavička posílá tak, jak je třeba, použil jsem analyzační nástroj Wireshark<sup>6</sup>, pomocí kterého jsem vysledoval TCP spojení mezi webovým serverem a dekodovacím softwarem. Na výpisu č. 6.2 je vidět výstup z analyzačního nástroje Wireshark.

---

```
1 GET /xml HTTP/1.1
2 Host: 192.168.4.1
3
4 HTTP/1.1 200 OK
5 Content-Type: text/xml; charset=utf-8
6 Connection: keep-alive
7
8 <?xml version="1.0" encoding="utf-8"?>
9 <telemetry>
10 <temp>19.30</temp>
11 <pressure>985.18</pressure>
12 <altitude>236.03</altitude>
13 <ground>1.53</ground>
14 </telemetry>
```

---

Výpis 6.2: HTTP hlavička a XML data

---

<sup>6</sup>Zdroj: <https://www.wireshark.org/>

Ve výpisu č. 6.2, je vidět na řádcích 1 a 2 dotaz dekódovacího softwaru na spojení s webovým serverem. V tomto případě, jak je vidět na řádce 1, je dotaz realizován dotazovací metodou GET s parametrem /xml (přístup na adresu 192.168.4.1/xml) a na řádce 2 je vidět IP adresa cíle, se kterým se chce dekódovací software spojit. Odpovědí na dotaz je pak HTTP hlavička, kterou vidíme na řádcích 4, 5 a 6. Důležitou součástí HTTP hlavičky je stavový kód 200, který říká, že HTTP požadavek byl úspěšný. Dalším důležitým polem HTTP hlavičky je i `Content-Type`, v tomto případě je nastaven na `text/xml`, což říká, že budou následovat XML data. Na řádcích 8 – 14 je už samotný XML dokument, stejný jako na výpisu č. 6.1.

Po úspěšné odpovědi, ve formě HTTP hlavičky a XML dat, je spojení mezi webovým serverem a dekódovacím softwarem ukončeno a webový server opět čeká na dotaz.

## 6.2 Implementace dekódovacího softwaru

Pro implementaci dekódovacího softwaru jsem se rozhodl použít programovací jazyk C#, jazyk vyvinutý firmou Microsoft. Tento programovací jazyk spadá do objektově orientovaných programovacích jazyků a je založen na jazycích C++ a Java.

Cílem dekódovacího softwaru je zobrazení telemetrických dat z telemetrického modulu v uživatelském rozhraní. Software umí zobrazit minimální a maximální naměřené hodnoty dané relace a výsledky celé relace uložit do tabulky. Uživatel pak s uloženými daty může dále pracovat.

### 6.2.1 Získání telemetrických dat z modulu

Navázání spojení dekódovacího softwaru s telemetrickým modulem je použita třída `WebClient`, pomocí které metodou `DownloadString()` získá software řetězec z adresy 192.168.4.1/xml, který obsahuje XML dokument. Tento řetězec se ovšem musí dále zpracovat, aby bylo možné vyčíst z něj uložené hodnoty. Pomocí třídy `XmlDocument` lze získaný řetězec převést na XML dokument, se kterým se poté dá pohodlně pracovat. Po získání XML dokumentu můžeme tento dokument rozdělit na jednotlivé uzly pomocí třídy `XmlNodeList`. Získané hodnoty z jednotlivých položek XML dokumentu lze uložit do proměnných `double`. Tyto proměnné jsou pak přidány, jako nová položka kolekce `List` (seznam). K jednotlivým položkám seznamu lze přistupovat pomocí indexu. Například pomocí `values[5].temp` se načte teplota měřená v šesté iteraci.

V implementovaném softwaru je ošetřena situace, kdy není navázání spojení s telemetrickým modulem úspěšné. V tomto případě je pomocí bloků `try` a `catch` odchycena výjimka a uživatel je upozorněn chybovou hláškou a možnými příčinami chyby.

Navázání spojení a načítání telemetrických dat je započato kliknutím uživatele na tlačítko „Connect“. Načítání telemetrických dat je v realizováno ve smyčce `while`, dokud uživatel neukončí spojení kliknutím na tlačítko „Disconnect“. Metoda `recieveData()`, která realizuje spojení s



modulem a získání telemetrických dat, je vykonána v odděleném vlákne tak, aby byla umožněna interakce uživatele se softwarem. Z tohoto vlákna ovšem není možné měnit prvky uživatelského rozhraní, proto je nutné použít metodu `Dispatcher.Invoke()`, pomocí které lze měnit data mimo vlákno.

Tlačítka „Connect“ a „Disconnect“ jsou realizována pomocí jediného tlačítka, které mění svou funkci v závislosti na situaci, ve které se software nachází. Funkce je řízena pomocí proměnné `bool`, která je při spojení s modulem nastavena na `false` a při odpojení od modulu se proměnná nastaví na `true`.

### 6.2.2 Zobrazení dat v uživatelském rozhraní

Pro zobrazení jednotlivých dat na obrazovce uživatele je využito prvků `Label`. Tyto prvky se v závislosti na změnách proměnných v kolekci `List` aktualizují. Jsou zobrazeny vždy poslední měřené hodnoty uložené v seznamu. U každé naměřené fyzikální veličiny je navíc zobrazena minimální a maximální naměřená hodnota dané relace. Porovnávání aktuální hodnoty s minimem a maximem je realizováno pomocí metody `Math.Min()` a `Math.Max()`. Mimo měřených veličin z modulu je v uživatelském rozhraní zobrazen i datum a čas aktuálního měření. Datum a čas je taktéž, spolu s měřeními telemetrickými daty, součástí seznamu.

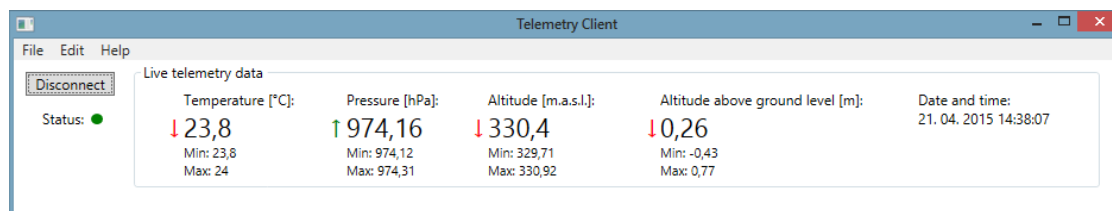
### 6.2.3 Uložení dat do CSV souboru

Veškerá měřená data je možné uložit do CSV souboru. Vytvoření CSV souboru probíhá pomocí metody `createCSV()`, která vrací hodnotu řetězce, kdy se při jejím volání předává do metody seznam s měřeními daty. Poté se projede celý seznam řádek po řádku a mezi jednotlivá data se vloží znak středníku, který zajišťuje oddělení buněk v CSV tabulce. Jelikož se celý řetězec s uloženými daty často mění – přidávají se do něj v každé iteraci nová data, byla použita instance třídy `StringBuilder()`, která zlepšuje výkonnost při připojování nových dat do řetězce.

Ještě před samotným navrácením řetězce, se do řetězce uloží hlavička popisující jednotlivé sloupce CSV tabulky. Takto zformátovaný řetězec se pomocí instance dialogu uložení `SaveFileDialog()` a statické metody `WriteAllText()` uloží do požadovaného CSV souboru s příponou `*.csv`. Takto uložený soubor lze dále otevřít ve většině softwaru pro práci s tabulkami, kdy z výstupních dat lze vytvářet grafy, dle libosti uživatele.

### 6.2.4 Použití dekódovacího softwaru

Ovládání softwaru je přímočaré a jednoduché. Jak je vidět na obrázku č. 6.2, software disponuje minimem ovládacích prvků, aby používání bylo pro uživatele co nejjednodušší.



Obrázek 6.2: Uživatelské rozhraní dekodovacího softwaru

První věc, kterou musí uživatel udělat před samotným spojením softwaru s telemetrickým modulem, je připojení se na WiFi přístupový bod s názvem „Telemetry\_Module“ heslem „vsbBEL0075“. Po úspěšném připojení k WiFi přístupovému bodu už nic nebrání ke sběru dat z telemetrického modulu. Pro připojení dekodovacího softwaru s telemetrickým modulem slouží tlačítko „Connect“, které je umístěno v levé části okna softwaru. Po kliknutí na toto tlačítko se změní stav tečky „Status“ z červené na oranžovou barvu, což značí, že se software snaží spojit s telemetrickým modulem. Po úspěšném spojení s telemetrickým modulem se tečka „Status“ změní na zelenou barvu a tlačítko „Connect“ dostává novou funkci a popisek „Disconnect“.

V pravé části okna jsou vyobrazena aktuálně měřená data – teplota, atmosférický tlak, nadmořská výška, výška nad úrovní země a také aktuální čas dekodování. Tyto hodnoty se aktualizují každých 1 - 2 sekund. Pod jednotlivými měřeními veličinami jsou zobrazeny také naměřená minima a maxima. Nalevo od každého měřeného údaje je vyobrazena šipka, která značí, zda se hodnota oproti minulému měření zvětšila či zmenšila. Pokud šipka zobrazena není, je aktuální hodnota stejná, jako ta předchozí.

Pro ukončení spojení slouží tlačítko „Disconnect“. Po kliknutí na něj, se změní tečka „Status“ zpět na červenou barvu – modul je odpojen od softwaru. Po odpojení je pak možné měřená data uložit či smazat a začít znovu s prázdným seznamem. Pro uložení stačí kliknout na File → Save. Po kliknutí na „Save“ se zobrazí dialogové okno s možností uložení CSV dokumentu s měřenými daty. Kliknutím na Edit → Erase measured data se měřená data vymažou.

Software je ošetřen i proti chybám vykonaným uživatelem. Při neúspěšném pokusu spojit se s telemetrickým modulem je uživatel informován chybovým oknem s doporučením, jak dále pokračovat. Pokud je seznam s měřenými daty prázdný, nelze kliknout na tlačítko pro uložení „Save“ a na tlačítko pro vymazání seznamu „Erase measured data“.

---

Date and time	Temperature [°C]	Pressure [hPa]	Altitude [m.a.s.l.]	Alt above ground level [m]
17. 4. 2015 14:42:19	21,6	966,32	399,09	0
17. 4. 2015 14:42:21	21,4	966,22	398,82	-0,27
17. 4. 2015 14:42:23	21,4	966,24	399,09	0
17. 4. 2015 14:42:26	21,4	966,24	398,23	-1,21

Tabulka 6.1: Ukázka výstupní tabulky

V tabulce č. 6.1 je vyobrazen příklad výstupu CSV dokumentu z dekodovacího softwaru. Tabulka je rozdělena do pěti sloupců, ve kterých jsou zaznamenána jednotlivá měřená data.

**Výhody:**

- Komplexní zobrazení dat
- Možnost uložení dat do CSV tabulky
- Rychlejší přenos

**Nevýhody:**

- Nutnost instalace dekodovacího softwaru

## 7 Testování

Testování telemetrického modulu proběhlo na kvadrokoptěře vlastní výroby. Použití kvadrokoptéry ovšem není podmínkou. Telemetrický modul se dá využít téměř u všech RC modelů, od kvadrokoptér po RC auta.

### Kvadrokoptéra

Kvadrokoptéra je vrtulník se čtyřmi rotory. V dnešní době jde o čím dál populárnější RC modely, které přichází v různých variantách. Můžeme se setkat také s označením *dron*, což je ovšem obecné označení pro bezpilotní letouny. Kvadrokoptéry se dnes využívají v amatérském modelářství, ale i pro účely policie, hasičů či armády.

Je důležité, aby kvadrokoptéra měla vhodné parametry pro použití s telemetrickým modulem. Jde zejména o nosnost, velikost a vhodné napájení (5 V a více). Použitá kvadrokoptéra pro testování telemetrického modulu je složena z dílů v tabulce č. 7.1.

Název	Popis	Parametry
X525	Hliníkový rám	Rozpětí 600 mm
D2826-10	Motory	1400 ot./min./V
HK Multi-Rotor Control Board V3.0	Řídící deska	Mikroprocesor Atmega328 PA
HK-HW30A	Regulátor otáček	Trvalý proud/max. 25/30 A
HK-TR6A v2	Přijímač	Pásmo 2,4 GHz
HK-T6A v2	Vysílač	Pásmo 2,4 GHz
2x levá vrtule, 2x pravá vrtule	Dvoulisté vrtule	254 mm × 11,4 mm
ZIPPY Flightmax 2200mAh 3S1P 40C	Baterie	Kapacita 2200 mAh

Tabulka 7.1: Parametry kvadrokoptéry

Aby nebylo nutné mít na kvadrokoptěře dvě baterie – jednu pro telemetrický modul a druhou pro napájení kvadrokoptéry, je možné využít stabilizátoru napětí v regulátoru otáček. Regulátory otáček pro jednotlivé motory jsou zapojeny do řídící desky. Řídící deska poskytuje možnost využít až šesti motorů (pro hexakoptéru), ale kvadrokoptéra využívá pouze čtyř motorů, je tedy možné využít zbylé piny pro napájení telemetrického modulu. Odebírané napětí z řídící desky je 5 V, což je dostačující k napájení mikropočítače Arduino Nano. Je nutné využít stabilizátoru napětí LM1085IT-3.3 na 3,3 V, pomocí kterého se může napájet senzor BMP180 a bezdrátový modul ESP8266.

## 7.1 Průběh testování

Po správném zapojení kvadrokoptéry a telemetrického modulu již nic nebrání tomu vyzkoušet telemetrický modul ve vzduchu. Při zapnutí kvadrokoptéry se zapne i telemetrický modul. Je nutné počkat asi deset sekund, aby se modul správně inicializoval. Po inicializaci je možné připojit se z notebooku nebo mobilního telefonu na WiFi přístupový bod s názvem „Telemetry\_Module“ heslem „vsbBEL0075“. Pokud chceme využít dekodovacího softwaru na notebooku, zapneme ho a připojíme se k modulu tlačítkem „Connect“. Pokud notebook nemáme, můžeme využít mobilního telefonu a v internetovém prohlížeči se připojit na adresu 192.168.4.1.

Nyní již jen stačí s kvadrokoptérou vzlétnout a sledovat průběh přenosu telemetrických dat na obrazovce počítače či mobilního telefonu. Po dokončení letu je v dekodovacím softwaru možné uložit naměřená data do CSV tabulky.

Na obrázku č. 7.1 je vidět kvadrokoptéra s telemetrickým modulem.

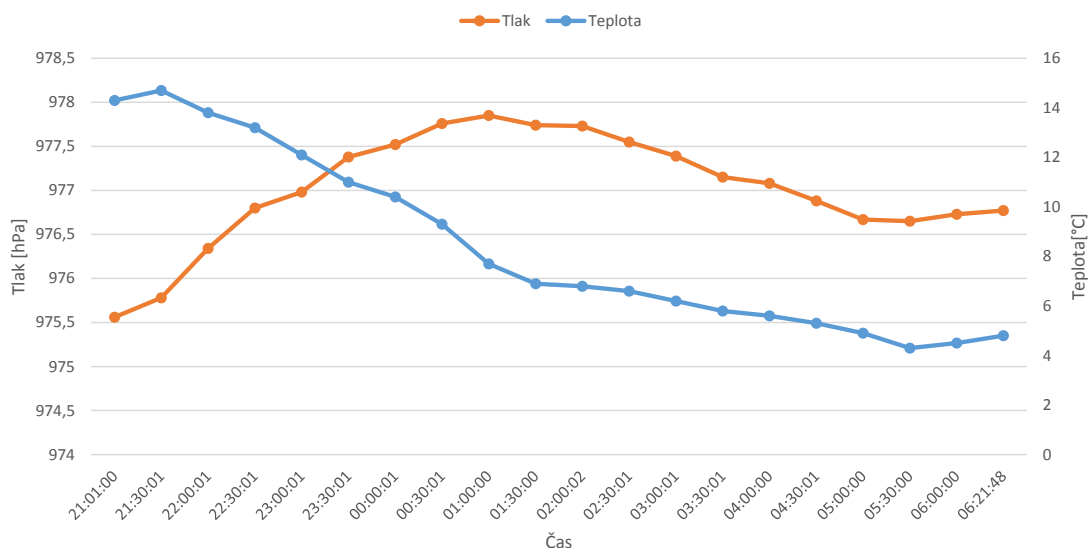


Obrázek 7.1: Kvadrokoptéra s telemetrickým modulem

## 7.2 Test stability

Abych si ověřil, že je modul stabilní a nevypadne komunikace po určité době, vykonal jsem test stability. Při tomto testu jsem nechal modul běžet přes noc venku. Test proběhl od 21.4.2015 21:00:58 do 22.4.2015 6:21:48 a za tuto dobu modul odeslal 16 043 záznamů do dekodovacího softwaru. Tento interval odpovídá 33 650 sekundám, z čehož můžeme jednoduše vypočítat průměrný interval příchodu nového záznamu do dekodovacího softwaru:  $33650 \div 16043 = 2,097s$ .

Pro zajímavost dokládám graf z měření, s teplotou a atmosférickým tlakem. Graf je vyobrazen na obrázku č. 7.2. Pro přehlednost je do grafu zaneseno měření v intervalu třiceti minut.



Obrázek 7.2: Měření teploty a tlaku

Vzhledem k těmto logovacím možnostem je možné využít modul i mimo pole RC modelů, třeba jako meteostanici.

### 7.3 Výsledky testování

Testování telemetrického modulu proběhlo, jak s použitím dekodovacího softwaru, tak s použitím internetového prohlížeče. Telemetrický modul i dekodovací software fungovaly dobře. Při přenosu nenastala chyba, všechny hodnoty se správně uložily a spojení bylo po celou dobu testovacího letu stabilní. Měření dosahu proběhlo s překvapivým výsledkem 430 m, což je velmi dobrý výsledek na tak malý modul, jakým je ESP8266.

Dalším měřeným parametrem je odběr proudu. Pokud je bráno v úvahu, že ATmega328P při 16 MHz má odběr proudu okolo 10 mA, senzor BMP180 má odběr proudu 12  $\mu$ A a modul ESP8266 má maximální odběr proudu 215 mA, vychází celkový odběr proudu zhruba na 241 mA, čemuž odpovídá příkon  $P = I \times U = 0,243 \times 5 = 1,205W$  [1, 10, 16].

Váha modulu je bez externí baterie 90 g, ovšem je nutné dodat, že se jedná o prototyp – to znamená, že velký podíl na hmotnosti má samotná deska nepájivého pole.

## 8 Závěr

Výsledkem této práce je realizovaný hardwarový telemetrický modul a dekodovací software pro zpracování dat. Vzhledem k měřeným údajům, jako je teplota, atmosférický tlak, nadmořská výška a výška nad úrovní země, lze telemetrický modul využít pro vědecké či meteorologické účely. Při návrhu telemetrického modulu jsem se snažil využít nové technologie a moderní moduly, mezi které patří modul ESP8266, který při svých malých rozměrech a nízké ceně poskytuje plnohodnotnou WiFi komunikaci a výborný dosah přes 400 m.

Základní funkcí dekodovacího softwaru je logování měřených dat a ukládání dat do souboru CSV, se kterým se dá dále pracovat ve většině tabulkových editorů. Oproti ostatním telemetrickým systémům se můj odlišuje ve využití modulu ESP8266, jako webového serveru a s tím spojenou možností zobrazení dat i v internetovém prohlížeči, bez nutnosti instalace jakéhokoliv softwaru.

Při psaní této práce jsem se dozvěděl mnoho nových a zajímavých informací z oblasti návrhu a realizace hardwarových modulů. Vzhledem k velkým možnostem v rozšíření se plánuji dále věnovat tomuto oboru a rozšířit telemetrický modul o další senzory, jako je například gyroskop či akcelerometr a vylepšit stávající dekodovací software o nové funkce.

Tato bakalářská práce mi poskytuje dobrý znalostní základ k dalšímu průběhu studia, případně při psaní diplomové práce.

## 9 Reference

- [1] ESP8266. *NURDs* [online]. 2013 [cit. 2014-10-10]. Dostupné z: <https://nurdspace.nl/ESP8266>
- [2] ESPRESSIF SMART CONNECTIVITY PLATFORM: ESP8266. In: *ESP8266 Specifications* [dokument ve formátu PDF]. 2013 [cit. 2014-10-10]. Dostupné z: [https://nurdspace.nl/images/e/e0/ESP8266\\_Specifications\\_English.pdf](https://nurdspace.nl/images/e/e0/ESP8266_Specifications_English.pdf)
- [3] NRF24L01+. *Nordic Semiconductor* [online]. [cit. 2014-10-13]. Dostupné z: <http://www.nordicsemi.com/eng/Products/2.4GHz-RF/nRF24L01P>
- [4] NRF24L01+ Single Chip 2.4GHz Transceiver. In: *nRF24L01+* [online]. 2008 [cit. 2014-10-23]. Dostupné z: [http://www.nordicsemi.com/eng/content/download/2726/34069/file/nRF24L01P\\_Product\\_Specification\\_1\\_0.pdf](http://www.nordicsemi.com/eng/content/download/2726/34069/file/nRF24L01P_Product_Specification_1_0.pdf)
- [5] Low - Power Sub - 1 GHz RF Transceiver. In: *CC1101* [online]. 2014 [cit. 2014-10-24]. Dostupné z: <http://www.ti.com/lit/ds/symlink/cc1101.pdf>
- [6] CC1101. *Texas Instruments* [online]. [cit. 2014-10-24]. Dostupné z: <http://www.ti.com/product/CC1101/description>
- [7] ISM band. *PC Mag Encyclopedia* [online]. [cit. 2015-01-24]. Dostupné z: <http://www.pcmag.com/encyclopedia/term/45467/ism-band>
- [8] Všeobecné oprávnění č. VO-R/10/05.2014-3 k využívání rádiových kmitočtů a k provozování zařízení krátkého dosahu. In: Praha, 2014. Dostupné z: [https://www.ctu.cz/cs/download/oop/rok\\_2014/vo-r\\_10-05\\_2014-03.pdf](https://www.ctu.cz/cs/download/oop/rok_2014/vo-r_10-05_2014-03.pdf)
- [9] Arduino Nano. *Arduino* [online]. [cit. 2015-02-03]. Dostupné z: <http://arduino.cc/en/Main/arduinoBoardNano>
- [10] BOSCH. *BMP180: Digital pressure sensor*. 2013 [cit. 2015-02-03]. Dostupné z: <https://ae-bst.resource.bosch.com/media/products/dokumente/bmp180/BST-BMP180-DS000-12.pdf>
- [11] *ESP8266 REFERENCE* [PDF]. 2014 [cit. 2015-3-15]. Dostupné z: <http://www.pighixxx.com/test/wp-content/uploads/2014/12/ESP8266Ref.pdf>
- [12] SCHERZ, Paul a Simon MONK. *Practical electronics for inventors*. Third edition. ISBN 978-007-1771-337.



- [13] RFC 2616 - Hypertext Transfer Protocol – HTTP/1.1 *IETF* [online]. 1999 [cit. 2015-04-26]. Dostupné z: <https://www.ietf.org/rfc/rfc2616.txt>
- [14] RFC 793 - Transmission Control Protocol. *IETF* [online]. 1981 [cit. 2015-04-26]. <https://www.ietf.org/rfc/rfc793.txt>
- [15] Introduction to XML. *w3schools.com* [cit. 2015-04-26]. Dostupné z: [http://www.w3schools.com/xml/xml\\_what\\_is.asp](http://www.w3schools.com/xml/xml_what_is.asp)
- [16] ATMEL CORPORATION. *ATmega48PA/88PA/168PA/328P*. 2009 [cit. 2015-04-28]. Dostupné z: <http://www.atmel.com/images/doc8161.pdf>